Refine Search

Search Results -

Terms	Documents
L26 same parity	22

Database: US Pre-Grant Publication Full-Text Database; US Patents Full-Text Database; US OCR Full-Text Database; EPO Abstracts Database; JPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: L27

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Tuesday, November 08, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L27	L26 same parity	22	L27
L26	L25 same information	40	L26
L25	L24 same wire	95	L25
L24	check adj1 bit	6687	L24
L23	L22 same horizontal	6	L23
L22	overall adj1 parity	162	L22
DB=USPT; PLUR=YES; OP=OR			
L21	overall adj1 parity	120	L21
L20	L13 same parity	8	L20
L19	horizontal same extended same parity	10	L19
L18	L15 same bit	32	L18
L17	L15 same symbol	0	L17
L16	L15 same wire	0	L16

L15	l8 same parity same error	34	L15
L14	L13 same bit same parity	5	L14
L13	wire same ecc	68	L13
L12	l8 near2 ecc	2	L12
L11	l8 and l10	0	L11
DB=USPT,PGPB; PLUR=YES; OP=OR			
L10	('4080649'| '5285451'| '5691994'| '5912906'| '6092231'| '6192499'| '6662334')![pn]	7	L10
DB=USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L9	L8 adj1 (error adj1 correction)	2	L9
L8	in-line	66583	L8
DB=USPT; PLUR=YES; OP=OR			
L7	L6 same parity	22	L7
L6	L4 same bit	47	L6
L5	L4 same wire	2	L5
L4	L3 same l2	179	L4
L3	(error or ecc) same (correction or detection)	102718	L3
L2	in-line	47817	L2
L1	5768296[pn] or 5787095[pn] or 6147827[pn] or 5157669[pn] or 5734663 [pn] or 6223309[pn] or 5517509[pn]	7	L1

END OF SEARCH HISTORY

L9: Entry 1 of 2 File: USPT Nov 1, 2005

DOCUMENT-IDENTIFIER: US 6961877 B2
TITLE: System and method for in-line error correction for storage systems

Abstract Text (1):
The present invention provides a method and system for performing in-line error correction in a disk storage system. The system includes an error correction (ECC) module; and a first memory storage device, wherein the first memory storage device and the error correction module simultaneously receive data from a storage disk before being buffered for transfer to a host system. The ECC module provides error correction mask before any data is transferred from the first memory storage device to a second memory buffer.

Brief Summary Text (3):
The present invention relates generally to disk controllers, and more particularly to in-line error correction before a host system transfers data to an external buffer during a read operation.

Brief Summary Text (14):
The present invention solves the foregoing drawbacks by providing a system for performing in-line error correction in a disk storage system. The system includes an error correction module; and a first memory storage device, wherein the first memory storage device and the error correction module simultaneously receive data from a storage disk before data is buffered for transfer to a host system. The ECC module provides error correction mask and the error is corrected before any data is transferred from the first memory storage device to a second memory buffer. The ECC module includes a module for generating error location and error mask information associated with data that is read from the storage device. The error location and error mask information is interleaved.

Drawing Description Text (16):
FIG. 9 is a block diagram showing various components used in in-line error correction, according to one aspect of the present invention; and

Detailed Description Text (20):
FIG. 2D shows various CH0 120 components with ECC module 109 components that are used for performing in-line error correction. CH0 control module 124 uses a queue control block 124, which is a part of ECC module 109 to access error mask 120E via error queue 134, MUX 162 (also shown in FIG. 9) and error mask register 162A. Plural commands between CH0 control module 124 are described below. As shown in FIG. 2E. Uncorrected data from FIFO 120B is XORed by WR data path module 120I with error mask 120E. Corrected data is then sent to buffer memory 111.

CLAIMS:

1. A system for performing in-line error correction during a read operation from a storage disk, comprising: a disk controller that includes a buffer controller coupled to a memory buffer, the memory buffer is external to the disk controller, wherein the buffer controller regulates data movement into and out of the memory buffer, and the buffer controller includes a first channel that interfaces with a disk formatter and the first channel includes a memory storage device; and an error

correction (ECC) module; wherein the memory storage device and the error correction module simultaneously receive data from the storage disk via the disk formatter and the ECC module generates error correction mask while data is still in the buffer controller and before any data is transferred from the memory storage device to the memory buffer.

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

End of Result Set

Generate Collection Print

L9: Entry 2 of 2 File: USPT Mar 1, 1988

DOCUMENT-IDENTIFIER: US 4729092 A
TITLE: Two store data storage apparatus having a prefetch system for the second store

Detailed Description Text (32):
However, as described above, there is a two-beat hold-up whenever a jump is made into or within the AMPS. Hence, the AMPS is preferably used for holding less frequently used microprogram sequences, or long sequences with no jumps. Also, because of the possibility of in-line error correction, the AMPS may be used to hold sequences for which roll-back would be undesirable or impossible.

Previous Doc Next Doc Go to Doc#

First Hit Previous Doc Next Doc Go to Doc#

Generate Collection Print

L23: Entry 3 of 6 File: JPAB Mar 3, 2005

PUB-NO: JP02005057741A
DOCUMENT-IDENTIFIER: JP 2005057741 A
TITLE: IN-LINE WIRE ERROR CORRECTION

PUBN-DATE: March 3, 2005

INVENTOR-INFORMATION:
NAME COUNTRY
DEBENDORA, DAS SHARUMA

ASSIGNEE-INFORMATION:
NAME COUNTRY
HEWLETT-PACKARD DEVELOPMENT CO LP

APPL-NO: JP2004205611
APPL-DATE: July 13, 2004

PRIORITY-DATA: 2003US-632908 (August 4, 2003)

INT-CL (IPC): H03 M 13/11; G06 F 11/10

ABSTRACT:

PROBLEM TO BE SOLVED: To provide an in-line wire error detection and correction scheme.

SOLUTION: In an in-line error detection and correction method using wires 0 to (k) and symbols 0 to (n), information bits and symbols are sent along the wires 0 to (k). Before sending an information block along the wires 0 to (k), check bits are calculated from the information bits, wherein the check bits are made up of horizontal parity, extended parity and overall parity of the information. The check bits are sent along the wires 0 to (k), using the same wires as for the information bits.

Previous Doc Next Doc Go to Doc#

L23: Entry 5 of 6 File: USOC Dec 7, 1965

DOCUMENT-IDENTIFIER: US 3222643 A
TITLE: Error detecting and correcting systems

OCR Scanned Text (17):
21 ing matrices. It is sometimes convenient, however, to tise other forms of lo-ical -ating iietworks. One such circuit is shown -'@n FIG. 7, and is infended for Lse with the coding described in Table 10. The locator subword signals kl-k4 are applied to a first grotip of recognition circuit-s 160-163, each of which prov;des an output signal only when a chosen signal pattern is present. Such recognition circuits are widely known and need not be descr-ibed in detail. Concurrently, the error type subword signals 7r, and 72 are applied to another group of recognition circuits 165-168. Associated AND gates 170-173, coupled to pairs (e.g. AND gate 170 is coupled to recognition circuits 160 and 165) of the recognition circuits, provide a signal through an OR gate 175 only when both recognition circuits are activated. This circuit therefore continually tests the signal patterns derived during shifting of the locator sequence generator 141,3 and the error type sequence generator 144 when the system is operating in the error correcting mode. There is coincidence of two signal patterns only once, at the proper increment of the S16-S30 interval. The co:incidence detection also separately indicates the type of error, so that a separate gating circuit may enter the proper correction into th-@ shift register 101. In a preferred arrangement, the advantages of m-seqliences are affectively used by a particularly simple set of decision networks as shown in the system of FIG. 8. This systeni is set to use the m- sequences of Table 10, and to correct three-bit-wide error bursts. The locator sequence generator 143 of FIG. 5 is coupled to provide signals to both a first decision network 146' and a second decision network 147'. The first network 146' contains a group of gates wbich serve an "exclusive OR" function, each gate being designated by a circle encompassing an intersection of two lines. An ini)ut signal on line kl, for example, provides an output sigiial on the D2 line through the gate 180, whereas an input signal on both lines k, and k4 results in an output signal only on line DI through the gate 181. Sig-iials applied to t@@e gates 180 and 182 on the D2 line are mutually inhibited oiid no outptit signal is derived. In effect, mod. 2 stims are- provided. The second decision network 147' is likewise coupl@ed to provide coded output signals in response to kl-k4Inspection of both of these networks 146' and 147' relative to Table 10 will reveal that the same function is performed as in the FIG. 7 circuit. That is, the first decision network 146' prov@.des an error correction pattern to the shift register 101, while a test is made of the relationship between the two subwords in the combination. To make the test, however, the kl-k,, conibination is converted to a 7rl--lr2 equivalent in the second decision network 147', and this ew@livalent is coti-ipared in the comparator 149 to the actual v, and 7r2 signals then prov,,ded from the error type sequence generator 144 of FIG. 5. The econoniy of this cirr-ii-it is derived froin advanta.-co@,is use of the properties of the m-sequences in arranging the matrices which perform the conversions. Those skilled in the art will reco,@nize that the principles of the invention may be extended to much niore complicated codes. As one example, it may be desired to use 31 bit messa.-Cs in which there are 20 infor@nation bits, 5 locator par-ity bits, 5 error type parity bits and I overall parity check bit. Usin.- the characteristic equation X5+X3+1=0 for the locator m-sequence, and the characteristic equation XI+X3+X2+X+1@o for the error type m-seqtience, the following sequences are obtained: 1 6 11 16 21 26 31 -- - - - --- Locatorsequence---- 00001 01011 10ilo

Golli 11001 10100 E,rrortypesequence- 00001 01101 01000 11101 11110 olool - 3,222,643 22 The shifts ar@d the parity for each error pattern are given here as Table I 1: Table 11 5 Shifts (Remainders modulo 31) Er ror pattern O verall S L S T S T- SL cli pek bit 10000 --------- -------- 0 0 0 1 10 10 001 --------- -------- 25 17 23 0 10 010 --------- -------- 5 8 3 0 10 011 --------- -------- 15 21 3 1 10 1(0 --------- ------- 28 2 4 27 0 10 101 --------- -------- 13 2 3 10 1 10 110 ---- ----- -------- 7 18 11 1 10 111 --------- -------- 9 2 2 13 0 11 000 --------- ---- ---- 14 12 29 0 15 11 001 --------- -------- 10 7 28 1 11 010 --------- -------- 26 10 15 1 11 011 --------- -------- 19 2 0 1 0 11 100 --------- -------- 22 2 7 5 1 11 101 --------- -------- 21 3 0 9 0 11 110 --------- -------- 11 5 25 0 --------- -------- 20 16 27 1 20 Note that ST-SL=3 occurs twice, as well as ST-SL=27 In both instances, however, the parity check bit given in 25 the last column can specify which of the two error patterns has occurred. One check pattern suitable for combining the information bits and parity bits into a suitable message grouping is as follows (Table 12): 30 Table 12 BIT NUMBER IN MESSAGE 1 6 11 16 21 20 31 - - 35 c --------- ---- 11111 '1' 11 11111 11 111 11111 lit il 1 P 4 --------- ---- 10101 11011 0 0011 11100 11010 01000 0 P 3 --------- ---- 01010 11101 10 001 11110 ol lot 00100 0 P 2 --------- ---- 00101 01110 11000 11111 0 0110 10010 0 Pi --------- ---- 00010 10111 01 1(0 01111 10 011 01001 0 D o --------- ---- 00001 01011 10110 00111 11001 10100 1 P4 --------- ---- 10110 10100 01110 11111 00100 11000 0 4.0 p ------- -- ---- 01011 01010 00111 01111 10010 01100 0 p -- ------ ------ 00101 10101 00011 10111 11091 00110 0 --------- ---- 00010 11010 10001 11011 11100 10011 0 Po ------- -- ---- 00001 01101 01000 11101 11110 01001 1 Inforiuatio-@i bits --------- 1 6 11 16 ---------- 20 ---- Check bits ---- ------- ----- - ------- C kik3k2kiko @4V3@211 @O 40 Error correcting means suitable for operating with this error correctin- code and message grouping are shown in FIG. 9. Here-again a circle about a crosspoint in a decision network means that the signal on the vertical through ro the crosspoint is one of the terms in a su:m modulo 2 wiiieh is defined by al@l the circled crosspoints on the horizontal. Elements like those of FIGS. 5 and 8 are similarly nun-lbered. In this arrangement, output signals froin the first deci- 55 sion network 181 are compared bit by bit to those from the error type sequence generator 144 and the overill check bit register 180 in a comparator 182. The comparator 182 includes modulo 2 adders and an OR circuit 184 to which is also coupled an inverter 133 (-Ierivin.- si.-- 60 nals from a second decision network 136 which generates the proper correction code for the error burst. Because the comparator 182 is set to provide reco-nition when the bit by bit comparisons are not all equal, @signals from the OR circuit 1:84 are applied to the correction co@itrol 65 gates 154 through an inverter 185. The step-by-step operation of this circuit is like that generally described in conjunction with FIG. 5. The comparison between the output signals from the first decision network 131 and the special line from t secon 70 decision network 186 with the signals froni the error type sequence generator 144 and the overall check bit register ;180 tests for concurrent identification of the same error pattern. The faulty burst of data in the shift register is c6neurrently shifted into the position at which it may be i5 corrected. Upon recognition of the error pattern the out-

Previous Doc Next Doc Go to Doc#